

```

//#####
//
// FILE:      LCD_20x4.c
//
// TITLE:     LCD Functions.
//
//#####
//
// Ver | dd mmm yyyy | Who | Description of changes
// =====|=====|=====|=====
// 1.00| 28 Mar 2006 | C.S. |
//#####

```

```
#include "DSP281x_Device.h"
```

```
// Pulse the enable pin high then low to execute an instruction
void pulse(void)
```

```
{
    Uint16 i;
    GpioDataRegs.GPADAT.bit.GPIOA1=1;
    for(i=0; i<3000; i++){asm(" nop");}
    GpioDataRegs.GPADAT.bit.GPIOA1=0;
    for(i=0; i<3000; i++){asm(" nop");}
}
```

```
void wait(Uint16 a)
```

```
{
    Uint16 i,j;
    for(i=0; i<60000; i++)
        {for(j=0; j<a; j++){asm("  nop");}}
}
```

```
// Initialize LCD
```

```
void init_lcd(void)
```

```
{
    Uint16 dat;
    GpioDataRegs.GPADAT.bit.GPIOA0=0;
    dat=0x003B;
    GpioDataRegs.GPADAT.all=dat<<2;
    pulse();
    dat=0x003B;
    GpioDataRegs.GPADAT.all=dat<<2;
    pulse();
    dat=0x000C;
    GpioDataRegs.GPADAT.all=dat<<2;
    pulse();
    dat=0x0001;
    GpioDataRegs.GPADAT.all=dat<<2;
    pulse();
    dat=0x0006;
    GpioDataRegs.GPADAT.all=dat<<2;
    pulse();
}
```

```
// Move cursor to column x, row y
```

```
void move_lcd(char x,char y)
```

```
{
    Uint16 dat,i;
    GpioDataRegs.GPADAT.bit.GPIOA0=0;
    for(i=0; i<300; i++){asm("  nop");}

    if (y == 0)
    {

```

```

        dat=(0x80 + x)<<2;
    }
    else if (y == 1)
    {
        dat=(0xC0 + x)<<2;
    }
    else if (y == 2)
    {
        dat=(0x94 + x)<<2;
    }
    else
    {
        dat=(0xd4 + x)<<2;
    }
    GpioDataRegs.GPADAT.all=dat;
    pulse();
}

// Write an ascii character to the current location
void write_lcd(char dat)
{
    GpioDataRegs.GPADAT.bit.GPIOA0=1;
    dat=(dat<<2)+1;
    GpioDataRegs.GPADAT.all=dat;
    pulse();
}

// Clear the LCD
void clear_lcd(void)
{
    Uint16 i;
    GpioDataRegs.GPADAT.bit.GPIOA0=0;
    GpioDataRegs.GPADAT.all=0x04;
    pulse();
    for(i=0; i<10000; i++){asm("    nop");}
}

// Draw a bar on column col (0-3) with length l (0-20)
void draw_bar(char row, char l)
{
    Uint16 i=0;
    move_lcd(0,row);

    if( l>20 || l<0)
    {
        write_lcd('e');
        write_lcd('r');
        write_lcd('r');
        write_lcd('o');
        write_lcd('r');
    }
    else
    {
        while(i < l)
        {
            write_lcd(0xFF);
            i++;
        }
        while(i<20)
        {
            write_lcd(' ');
            i++;
        }
    }
}

```

